REMARKS

Claims 1-3, 5-8 and 10-16 are pending in the present application. Claims 1 and 8 were amended to clarify differences between the present invention and the cited art and to incorporate subject matter previously recited in dependent claims 4 and 9, respectively; and claims 4 and 9 have been canceled. New claim 16 was added to more fully protect Applicant's invention.

Applicant has carefully considered the cited art and the Examiner's comments, and believes claims 1-3, 5-8 and 10-16 patentably distinguish over the cited art and are allowable in their present form. Reconsideration of the rejection is, accordingly, respectfully requested in view of the above amendments and the following comments.

I. 35 U.S.C. § 103, Obviousness

The Examiner has rejected claims 1-15 under 35 U.S.C. § 103(a) as being unpatentable over Applicant Admitted Prior Art (AAPA) in view of Gaither (U.S. Patent No. 6,223,256 B1). This rejection is respectfully traversed.

In rejecting the claims, the Examiner states as follows with respect to claim 1:

> As per claims 1, and 6-7, AAPA discloses a multiprocessor system comprising: a plurality of processors **[CPU0-CPU7 (Fig. 1)]**; a plurality of memories associated with and residing on a chip with at least some of said plurality of processors **[MEM0-MEM15 (Fig.1)]**; and a software-administered cache located in said plurality of memories **[software cache (paragraph 0005)]**, said software-administered cache is administered by an AIX operating system and said region is a segment **[semented architecture, such as Advanced Interactive eXective (AIX). AIX is a version of UNIX (paragraph 0010)]**, and said cache being separated into a plurality of cache classes **[the cache can be separated into a number of cache classes (paragraph 0005)]**.
>
> AAPA teaches the claimed invention, but fails to specifically disclose that cache memory associated with each of said cache classes is located within a single region within one of said memories; and wherein a first thread, performing at least one administrative function within said cache, is connected to operate sequentially.
>
> Gaither discloses storing of sequential data streams and sequential instruction streams into a cache memory. Gaither further discloses the cache may be logically partitioned into sections and sections may be dedicated to a corresponding classes in order to improve the cache performance by assigning different replacement algorithm for different classes of item that has been cached **[column 1, lines 51-52; column 4, lines 41-30]**.
>
> Accordingly, it would have been obvious to one having ordinary skill in the art at the time of the current invention to use cache partitioning and assignment as being taught by Gaither into AAPA's caching system in order to be able to assign different replacement algorithm to different sections of the cache, so that the cache performance will be improved.

BEST AVAILABLE COPY

Office Action dated December 19, 2005, pages 2-3.

In addition, with respect to claim 4, the subject matter of which has been incorporated into claim 1, the Examiner further states:

> As per claim 4, Gaither discloses that each of said plurality of cache classes is allocated a plurality of pages in contiguous memory [**see Fig. 1, 3A and 3B**].

Office Action dated December 19, 2005, page 4.

Claim 1 of the present application as amended herein is as follows:

> 1. A multiprocessor system comprising:
> a plurality of processors;
> a plurality of memories associated with and residing on a chip with at least some of said plurality of processors; and
> a software-administered cache located in said plurality of memories, said cache being separated into a plurality of cache classes, cache memory associated with each of said cache classes being located within a single region of a plurality of regions within one of said memories, and each of said plurality of cache classes being allocated a plurality of pages in contiguous memory,
> wherein a first thread, performing at least one administrative function within said cache, is connected to operate sequentially through the single region.

BEST AVAILABLE COPY

A fundamental notion of patent law is the concept that invention lies in the new combination of old elements. Therefore, a rule that every invention could be rejected as obvious by merely locating each element of the invention in the prior art and combining the references to formulate an obviousness rejection is inconsistent with the very nature of "invention." Consequently, a rule exists that a combination of references made to establish a *prima facie* case of obviousness must be supported by some teaching, suggestion, or incentive contained in the prior art which would have led one of ordinary skill in the art to make the claimed invention.

The Examiner bears the burden of establishing a *prima facie* case of obviousness based on the prior art when rejecting claims under 35 U.S.C. § 103. *In re Fritch*, 972 F.2d 1260, 23 U.S.P.Q.2d 1780 (Fed. Cir. 1992). The requirements for establishing a *prima facie* case of obviousness in view of a combination of references are set forth in detail in Section 2142 of the MPEP and include the requirements that the Examiner explain in detail why the combination of the teachings is proper, that the Examiner provide a clear and convincing line of reasoning as to why an artisan would have found the claimed invention obvious in light of the teachings of the references, and that the Examiner provide a showing that it is the prior art and not the Applicant's own disclosure that teaches the combination asserted by the Examiner.

Neither AAPA nor Gaither nor their combination discloses or suggests "a software-administered cache located in said plurality of memories, said cache being separated into a plurality of cache classes, cache memory associated with each of said cache classes being located within a single region of a plurality of regions within one of said memories, and each of said plurality of cache classes being allocated a plurality of pages in contiguous memory" and "wherein a first thread, performing at least one administrative function within said cache, is connected to operate sequentially through the single region" as recited in claim 1; and claim 1 is, therefore, not obvious in view of AAPA and Gaither.

Gaither is directed to a cache memory system for a computer. In Gaither, target entries for the cache memory include a class attribute, and the cache may be partitioned into sections based on class attributes. Gaither further discloses that different classes of items being cached have a different replacement algorithm to improve cache performance.

Col. 4, lines 14-32 of Gaither, referred to by the Examiner in rejecting the claims, reads as follows:

> There is need for further improvements in cache performance for data streams having mixed locality. There is a need for further improvements in providing partial cache locking. In particular, there is a need for a cache having different replacement algorithms for different classes of items being cached.
>
> SUMMARY OF THE INVENTION
>
> Multiple example embodiments of cache systems are disclosed that provide a great deal of flexibility in optimizing cache performance for a wide variety of applications. In each example embodiment, items to be cached have a corresponding class attribute. Each class attribute may have a different corresponding replacement algorithm. A cache may be logically partitioned into sections. Cache logical sections may be dedicated to corresponding classes or the sections and classes may be ranked in a hierarchy with some sections corresponding to multiple classes.

In the above recitation, Gaither discloses that a cache may be logically partitioned into sections, with each section dedicated to corresponding classes or that the sections and classes may be ranked in a hierarchy, and that a different algorithm be provided for different classes of items being cached. Gaither does not, however, disclose or suggest in the above recitation or elsewhere in the patent that that in a multiprocessor system having a plurality of memories associated with and residing on a chip, and a cache located in the plurality of memories and separated into a plurality of cache classes, a "cache memory associated with each of said cache classes being located within a single region of a plurality of regions within one of said memories" (emphasis added) as recited in claim 1. The statement in Gaither that a cache may be logically partitioned into sections with each section dedicated to a class is, in no way, a teaching or suggestion of "a software-administered cache located in said plurality of memories, said cache being separated into a plurality of cache classes, cache memory associated with each of said cache classes being located within a single region of a plurality of regions within one of said memories, and each of said plurality of cache classes being allocated a plurality of pages in contiguous memory" and "wherein a first thread, performing at least one administrative function within said cache, is connected to operate sequentially through the single region" as now recited in claim 1.

As pointed out in the present specification, by completely containing a cache class within a single region of a plurality of regions of a memory of a plurality of memories, whenever a least recently used page must be located, or the pages synchronized, the thread can pass through the allocated region sequentially with reduced overhead (see, for example, the first paragraph on page 8 of the present specification).

In rejecting claim 4 (the subject matter of which is now incorporated in claim 1), the Examiner referred to Fig. 1 and Figs.3A and 3B of Gaither. Figs. 3A and 3B, Gaither are described in col. 9, lines 11-35 as follows:

> FIGS. 3A and 3B illustrate still another example embodiment of the invention, a TLB partitioned into classes. Recall that the TLB is also a cache. Therefore, the TLB may be partitioned into sections corresponding to classes. Note that either paged or segmented systems may include a TLB or its functional equivalent but in the specific example of FIGS. 3A and 3B paging is assumed. In FIGS. 3A and 3B, the TLB is a fully associative (one set) cache and the set is partitioned into three sections (300, 302, 304).
>
> So far, in the discussion of FIGS. 1 and 2, it has been assumed that the section placement algorithm dedicates each section to a single attribute class. For example, in FIG. 3A, Section 0 (300) may be dedicated to Class A only (306), Section 1 (302) may be dedicated to Class B only (308) and Section 2 (304) may be dedicated to Class C only (310). As a result, part of the cache (for example, Section 2) is unavailable to items that may have a high degree of locality (for example, class A). As an alternative, the classes may be ranked in a hierarchy, with Class A having a higher rank in the hierarchy than Class B and so forth. In FIG. 3A, this is illustrated on the right-hand side where Class A items may be placed anywhere in the cache, Class B items may be placed in Section 1 (302) or into any section corresponding to a class having a lower rank in the hierarchy than Class B. Finally, Class C items may be placed only in Section 2 (304).

Although Gaither may disclose that a TLB (Translation Look-Aside Buffer) may be partitioned into classes, this portion of Gaither also does not disclose or suggest "a software-administered cache located in said plurality of memories, said cache being separated into a plurality of cache classes, cache memory associated with each of said cache classes being located within a single region of a plurality of regions within one of said memories, and each of said plurality of cache classes being allocated a plurality of pages in contiguous memory" as now recited in claim 1.

Thus, neither AAPA nor Gaither discloses or suggests "a software-administered cache located in said plurality of memories, said cache being separated into a plurality of cache classes, cache memory associated with each of said cache classes being located within a single region of a plurality of regions within one of said memories, and each of said plurality of cache classes being allocated a plurality of pages in contiguous memory" and "wherein a first thread, performing at least one administrative function within said cache, is connected to operate sequentially through the single region", as recited in claim 1, and in the absence of any such disclosure in either reference, one of ordinary skill in the art would not find the claimed invention to be obvious in light of the teachings of the references. Accordingly, the Examiner has not established a *prima facie* case of obviousness in rejecting claim 1 over AAPA in view of Gaither, and claim 1 patentably distinguishes over the references and should be allowable in its present form.

Claims 2-3 and 5-7 depend from and further restrict claim 1, and are also not obvious over AAPA in view of Gaither, at least by virtue of their dependency.

Independent claim 8 has been amended in a manner similar to claim 1, and is also not obvious over AAPA and Gaither for substantially the same reasons as discussed above with respect to claim 1. Claim 8, accordingly, together with claims 10-15 dependent thereon should also be allowable over the references in their present form.

New computer program product claim 16 has been added to more fully protect Applicant's invention. Claim 16 corresponds to method claim 8 as amended herein, and should also be allowable in its present form for substantially the same reasons discussed above with respect to claim 1.

Therefore, the rejection of claims 1-15 under 35 U.S.C. § 103 has been overcome.

II. Conclusion

For all the above reasons, it is respectfully urged that claims 1-3, 5-8 and 10-16 are allowable in their present form, and that this application is now in condition for allowance. It is, accordingly, respectfully requested that the Examiner so find and issue a Notice of Allowance in due course.

The Examiner is invited to call the undersigned at the below-listed telephone number if in the opinion of the Examiner such a telephone conference would expedite or aid the prosecution and examination of this application.

DATE: March 20, 2006

Respectfully submitted,

Gerald H. Glanzman

Gerald H. Glanzman
Reg. No. 25,035
Yee & Associates, P.C.
P.O. Box 802333
Dallas, TX 75380
(972) 385-8777
Attorney for Applicant